Respuesta a preguntas:

¿Qué diferencia existe entre una arquitectura Data Warehouse y un Data Lakehouse? ¿Qué tipo de arquitectura le recomendaría a WWI para complementar lo desarrollado en este laboratorio incluyendo fuentes de datos no estructuradas, análisis en tiempo real que incluyan simultáneamente tanto datos estructurados como no estructurados?

La principal diferencia entre un Data Warehouse y un Data Lakehouse se basa en el tipo de datos que pueden manejar y la flexibilidad de su arquitectura resultante.

Un Data Warehouse tradicional está optimizado para el almacenamiento y procesamiento de datos estructurados, generalmente de sistemas transaccionales. Los datos se cargan en un esquema predefinido y rígido y se utilizan principalmente para análisis históricos y reportes.

Por otro lado, un Data Lakehouse es una arquitectura más moderna que combina las capacidades de un Data Lake (almacenamiento de datos sin procesar en su formato nativo, estructurado o no estructurado) con las funcionalidades de un Data Warehouse (almacenamiento estructurado, procesamiento optimizado y herramientas analíticas).

Un Data Lakehouse brindaría a WWI una arquitectura más flexible, escalable y adaptada a los requisitos actuales de análisis de datos, permitiendo aprovechar al máximo sus fuentes de datos estructuradas y no estructuradas.

¿Qué ventajas y desventajas observa al momento de implementar un ETL utilizando este tipo de herramientas respecto a desarrollarlo utilizando Python, Pandas y demás herramientas vistas durante la primera parte del curso?

De ventajas encontramos que GCP provee una infraestructura flexible, fácilmente adaptable al cambio en la demanda y el volumen de datos.

Estas soluciones al correr en la nube no son dependientes del equipo de cómputo, permitiendo que todo el proceso pueda realizarse por cualquiera sin importar la maquina donde se trabaje.

Adicionalmente, las herramientas existentes en GCP como DataPrep permiten hacer un trabajo de revisión y edición de pasos en el ETL. Estas herramientas facilitan bastante el trabajo permitiendo visualizar el flujo de datos, las

transformaciones y en general una experiencia más amigable trabajando con el ETL.

Finalmente, los servicios están integrados y testeados por todo un grupo de desarrolladores de estas herramientas. Lo que facilita el procesamiento, desarrollo y confianza.

Sin embargo, a diferencia de Python y las otras herramientas es una plataforma de pago. Los servicios al residir en la nube, es requerida también una conexión estable a internet. Finalmente, y relacionado con el anterior punto, cuando se trabaja con herramientas locales, se tiene total agencia de los datos y posibilidad de trabajar sin estar en una red. Pueden existir datos sensibles que se prefieran trabajar en un entorno más controlado, en tierra y sin conexiones.

¿Qué tipo de esquema, estrella o copo de nieve, representa el modelo multidimensional construido en este laboratorio? Justifica tu respuesta.

El modelo multidimensional construido en el laboratorio representa un esquema de estrella. Esto gracias a la presencia de una tabla de hechos central llamada OrderLine, que está directamente conectada a las tablas de dimensiones Date, PackageType, y StockItem, sin tablas de dimensiones adicionales interconectadas. Cada tabla de dimensiones contiene atributos específicos y la tabla de hechos central incluye claves foráneas que enlazan con las dimensiones, así como medidas como totalprice y quantity. Este diseño permite consultas eficientes y un modelo de datos simplificado, ideal para análisis y reportes.

¿Qué tipo de tablas de hechos y de medidas se identifican en el modelo multidimensional dado? Justifica tu respuesta.

Datasets Parametrizados: Se indican conjuntos de datos parametrizados como OrderLines, StockItems y PackageTypes, que podrían ser tablas de hajechos, ya que contienen datos transaccionales o eventos que se pueden medir y analizar.

Recetas de Datos: Las recetas como Filter, Select, Drop y Generate sugieren operaciones para transformar y preparar los datos.

Columnas Específicas: La selección y filtrado de columnas específicas, como OrderDate y StockItems ID, indica la identificación de medidas clave y dimensiones dentro de las tablas de hechos.

Outputs Generados: Los outputs como Generate Dates ID y FilterTypesIDAndNameColumns implican la creación de identificadores únicos y la filtración de información.

Suponga que la dimensión StockItemDim cambia el manejo de la historia de tamaño y precio del producto a un tipo 2 (Slow Change Dimension). ¿Qué ajustes a la dimensión relacionada con el producto, a la tabla de hechos y al proceso ETL se deben realizar para que al cargar la información se incluya este manejo de historia?

Para ajustar la dimensión StockItemDim a un tipo 2 de Slow Change Dimension (SCD), que maneja cambios históricos en los datos, se deben realizar los siguientes cambios:

Dimensión StockItemDim:

Agregar Campos de Historia: Incluir campos para la fecha de inicio y fin de la validez de cada registro, así como un indicador de registro actual.

Manejo de Llaves: Implementar llave primaria que permita múltiples entradas para el mismo producto con diferentes periodos de validez.

Tabla de Hechos:

Llave Foránea: Asegurarse de que la tabla de hechos referencie la llave primaria de la dimensión StockItemDim para mantener la integridad referencial.

Historia de Transacciones: Permitir que las transacciones se relacionen con la versión correcta del producto basándose en la fecha de la transacción para referenciar al elemento actual.

Proceso ETL:

Detección de Cambios: Modificar el proceso ETL para detectar cambios en tamaño y precio, y cargar una nueva entrada en la dimensión StockItemDim cuando ocurran.

Carga de Datos: Implementar la lógica para cargar datos históricos correctamente, incluyendo la actualización de las fechas de validez y el indicador de registro actual.

¿Qué errores se le presentaron en el desarrollo del laboratorio y qué solución plantearon? Haga énfasis en los que fueron más difíciles de solucionar.

De igual manera se tuvo problemas con la plataforma ya que a medida que se hacia el laboratorio con el uso de los videos y la guía a veces no era orden el claro los pasos a seguir, las explicaciones son un tanto ambiguas y encontrar algunas funciones no era tan trivial. El orden de los contenidos y su estructura podrían ser más claras.

Muchos de los procesos que se esperaba no serían tan arduas igual resultaban siéndolo y con más trabajo manual. Esto en particular hacia que dichos procesos fueran más propensos a errores.

Además de estos errores, hubo problemas sobre trabajar con SQL en GCP, no fue claro donde escribir y trabajar con las consultas en primera instancia. Finalmente, se encontró en el BigQuery la opción para trabajarlos.